

**REMARKS**

Claims 1-14 are presently in the application.

New Matter

The Examiner has indicated that the amendment submitted on March 3, 2011 is improper in that it incorporates new matter. More specifically, the Examiner alleges that there is no support in the description for the limitation "initiating one at a time" found in claim 1 as amended.

Support for the limitation "initiating one at a time" can be found throughout the description as originally filed. With regards to figure 2, the following is stated:

*[0010] Yet according to a further aspect of the invention, there is provided, in a computer system, a driver for providing improved real time command execution in a non real time operating system, comprising a command queue comprising a sequence of asynchronous commands to be handled in real time, a command dispatcher operating at a kernel mode level and providing a command of the sequence of asynchronous commands to a target unit in response to an "end of command execution" signal generated in said computer system. (emphasis added)*

From this passage, it should be understood that the command dispatcher provides a single command at a time from the sequence of commands.

*[0031] In the preferred embodiment, as illustrated by the timeline in FIG. 1, a user mode application calls the kernel with a sequence of commands to be executed. Once in kernel mode, the command dispatcher 4 stores commands in a command queue 6. The first command will be issued and executed. When the command is finish executing, a "command complete" signal will announce to the command dispatcher 4 that the hardware 7 is available to execute another command. After another small delay of handling the signal received, known as interrupt handling time, the command dispatcher 4 will issue another command from the queue 6. The process is repeated until there are no more commands in the command queue 6. (emphasis added)*

From this passage, it should be understood that commands are issued one at a time for execution.

*[0050] The command dispatcher 4 provides the next command of the sequence of commands to a corresponding target unit 8 in response to a reception of a "command executed" signal provided by the target unit 8. (emphasis added)*

From this passage, it should also be understood that the commands are provided one at a time. With regards to figure 6, the following is stated:

*[0077] If the command is first in the queue, then a request will be made by the command dispatcher 4 to provide the command to the appropriate target unit, as per step 54. (emphasis added)*

*[0079] In the case where the target unit 8 accepts the request for the command and according to step 58, the command is set to be executed by the target unit 8. Then, according to step 60, the system will wait for the "command executed" event, such as an interrupt. Following such an event, the executed command will be removed from the queue, as per step 62 and the algorithm will end. (emphasis added)*

These passages clearly indicate that commands are provided one at a time for execution. With regards to figure 7, the following is stated:

*[0088] According to steps 80 and 82, the command dispatcher scans each command queue in order of priority, looking for commands in pending mode. These are commands which have requested the target unit 8, while this was being used to execute a different command. Therefore, these are commands which were "ready" to execute but have been denied the hardware. Of all pending commands, the one with the highest priority will be the next one to be dispatched. (emphasis added)*

From this passage, it should be understood that commands are dispatched one at a time. In addition, the Applicant respectfully submits that the expressions “provides”, “issues”, “sets” and “dispatche[s]” are sufficient support for the term “initiating” found in claim 1. There is clear support in the description for the at least one CPU “initiating one at a time” execution of each of the commands from the stored sequence of commands.

The Applicant respectfully requests that the rejection be withdrawn.

#### Claim Rejections – 35 USC § 102

Claims 1-2 are rejected under 35 USC 102(e) as being anticipated by Baertsch et al. (US Patent No. 6,470,071) This rejection is respectfully traversed for the following reasons.

The Applicant understands that the arguments presented in the response of March 3, 2011 were not considered by the Examiner due to the concern regarding new subject matter. In light of the above, the Examiner is asked to reconsider the rejection to claims 1-2 on the basis of Baertsch et al. The previously provided arguments are reproduced herein for the Examiner’s convenience.

Claim 1, as amended, clearly indicates that the sequence of commands, provided to the privileged mode and to be executed in real time, is provided to at least one CPU running the non real time operating system, and that the commands are initiated one at a time, by the at least one CPU, for execution. In contrast, Baertsch et al. describe storing the commands on a hardware device (the Detector Framing Node (DFN)) and having the hardware device execute the sequence of commands in batch autonomously after the Host sends a “begin” command. This is evidenced by the passage found in column 14 at lines 48-54, where it states that “Event sequence 312 is initiated by sending a Begin Sequence command over computer communication bus 302. The extent of real-time control allotted to host computer 114 is confined to a determination of when event sequence 312 will begin. Subsequently, host computer 114 is completely removed from image acquisition”. It is important to note the distinction between the DFN device driver and the DFN hardware device in Baertsch et al. The device driver is only a tunnel in the operating system to send the list of commands in advance and write them in the hardware for later execution. Therefore, it should be understood that Baertsch et al. fail to teach or suggest at least the steps of “providing from said at least one application said sequence of commands to a privileged mode of said non real time operating system to be executed in real time” and “initiating one at a time, using the at least one CPU, execution of each of said commands from said stored sequence of commands”.

The Applicant respectfully requests that the rejection against claims 1-2 be withdrawn.

#### Claim Rejections – 35 USC § 103

Claims 3-14 are rejected under 35 USC 103(a) as being unpatentable over Baertsch et al. in view of Dingwall et al. (US 5,903,752).

Dingwall et al. does not address the deficiencies of Baertsch provided above. In addition, the statement in Baertsch regarding the lack of involvement of the host computer CPU with respect to image acquisition is a clear teaching away from any modification that would have the host computer CPU perform the steps recited in the present claims. For at least these reasons, the Applicant respectfully requests that the rejection of claims 3-14 be withdrawn.

#### Conclusions

It is believed that Claims 1-14 are allowable over the prior art and a Notice of Allowance is earnestly solicited.

Respectfully submitted,

Michel Doyon et al.

By:

/ALEXANDRA DAOUD/

Alexandra Daoud

Registration No 55,992

Customer Number: 020988